RAPPORT DE RECHERCHE

L R I

ENERGY CONSERVATION IN WIRELESS SENSOR NETWORKS AND CONNECTIVITY OF GRAPHS

LI H / MIAO H / LIU L / LI L / ZHANG H

Unité Mixte de Recherche 8623
CNRS-Université Paris Sud – LRI

12/2007

Rapport de Recherche N° 1482

CNRS – Université de Paris Sud
Centre d'Orsay
LABORATOIRE DE RECHERCHE EN INFORMATIQUE
Bâtiment 490
91405 ORSAY Cedex (France)

Energy Conservation in Wireless Sensor Networks and Connectivity of Graphs

Hao Li [a,c,1], Huifang Miao [a,b,*], Li Liu [a,c], Lian Li [d], Heping Zhang [c]

[a] *LRI, Univ Paris-sud and CNRS, Orsay F-91405, France*

[b] *School of Mathematical Sciences, Xiamen University, Xiamen Fujian 361005, P. R. China*

[c] *School of Mathematics and Statistics, Lanzhou University, Lanzhou 730000, P. R. China*

[d] *School of Information Science and Technology, Lanzhou University, Lanzhou 730000, P. R. China*

Abstract

In wireless sensor networks (WSNs), energy source is usually battery cell, which is impossible to recharge while WSNs are working. Therefore, one of the main issues in wireless sensor networks is how to prolong the network lifetime of WSNs with certain energy source as well as how to maintain coverage and connectivity. In this paper, we consider the wireless sensor networks satisfying that each node monitors one target or just for connection. Assume the wireless sensor network has l targets, and each is monitored by k sensor nodes. If $k = 2$ and the graph G corresponding to the wireless sensor network is $(l + \max\{1, l - 4\})$-connected, or $k \geq 3$ and G is $(l(k - 1) + 1)$-connected, then we can find k (the maximum number) disjoint sets each of which completely covers all the targets and remains connected to one of the central processing nodes. The disjoint sets are activated successively, and only the sensor nodes from the active set are responsible for monitoring the targets and connectivity, all other nodes are into a sleep mode. And we also give the related algorithms to find the k disjoint sets.

Key words: wireless sensor network, graph, connectivity.

* Corresponding author.
Email address: hfmiaomath@gmail.com (Huifang Miao).

1 The work was partially supported by NNSF of China (60373012)

1 Introduction

A wireless sensor network (WSN) is a wireless network comprised of a large number of sensor nodes which are deployed randomly. Wireless sensor networks have attracted a good deal of research attention as they used in many application areas, including battlefield surveillance, environment and habitat monitoring, home automation, inventory tracking, and healthcare application (11).

The energy source of a node is most often a battery cell, and they can stay active for a limited time before the battery resources are depleted. As recharging the battery is not feasible in many applications, energy efficient coverage is an important issue in wireless sensor networks design. In energy efficient coverage problem, the goal is to monitor the set of targets (or the area) with low energy consumption. Especially, in area coverage problem, the objective of wireless sensor network is to cover an area. While in point coverage problem, the goal is to cover a set of targets. Another important issue of wireless sensor networks is connectivity.

In this paper we address a special point coverage, which need to monitor a set of discrete targets with known locations. The sensor nodes are dispersed to send the monitored information to one or more central proceeding nodes. The most remarkable characteristics is a large number of sensor nodes are dispersed randomly in close proximity around each target for surveillance and other nodes only for connection, which implies each target is monitored by lots of sensor nodes.

In order to extent the network lifetime, organize the sensor nodes into disjoint sets each of which completely covers all the targets and remains connected to one of the central processing nodes, with only one set performing environmental monitoring and connectivity at any moment. These disjoint sets are activated successively. Scheduling and grouping of sensor nodes into disjoint sets is done by the central processing nodes and the synchronizer, which inform every sensor node to be activated or not. All sensor nodes of the active set are in the active state, whereas all other nodes are into a sleep state, where the CPU is in a low power mode and radio reception is disable. The ratio of energy consumed between the sleep state and the active state (i.e., when the CPU operates at full energy) is typically on the order of 100 or more (6). The goal is to maximize the number of disjoint sets each of which completely covers all the targets and remains connected to one of the central processing nodes.

Refer to (1) for graph theory notation and terminology not described here. Model the wireless sensor network with n sensor nodes as an undirected graph

G with n vertices. An edge exists between vertices u and v if and only if nodes u and v are within each other's sensing range. Usually, call G as a network graph. Call the vertices corresponding to the central processing nodes as central vertices. Assume the sensor network has l targets, and each is monitored by k sensors. For $1 \leq i \leq l$, let A_i be the vertex subset of G corresponding to the set of sensors monitoring each target, S the set of the central processing nodes, $A_i \cap S = \emptyset$. Moreover, for $1 \leq i < j \leq l$, $A_i \cap A_j = \emptyset$. A connected subgraph of G, which contains at least one vertex of S and one vertex of A_i for each i with $1 \leq i \leq l$, is corresponding to a set of sensor nodes which monitors all the targets and transmits the information to at least one of the central processing nodes. Hence, the maximum number of disjoint sets of the wireless sensor network is corresponding to the maximum number of such disjoint connected subgraphs.

The problem in wireless sensor networks containing more than one central processing nodes can be changed into the case in which there is only one central processing node. In fact, let G be the graph corresponding to the wireless sensor network which has more than one central processing nodes, and G' obtained by contracting the set of central vertices of G into a single vertex s. The connected subgraph of G containing at least one central vertex is corresponding to the connected subgraph of G' containing s. Hence, we only need to consider the wireless sensor networks with unique central processing node. In this paper, we find a parameter – connectivity of the network graphs to achieve energy conservation and connectivity.

Theorem 1 *Let G be a graph, $A_1, \ldots, A_l$ be any l pairwise disjoint vertex subsets with $|A_i| = k$ for $1 \leq i \leq l$, $s \in V(G) \setminus \cup_{i=1}^{l} A_i$. If $k = 2$ and G is $(l + \max\{1, l-4\})$-connected, or $k \geq 3$ and G is $(l(k-1)+1)$-connected, then there exist k connected subgraphs $G_1, \ldots, G_k$ such that*

(a) $|V(G_i) \cap A_j| = 1$ for $1 \leq i \leq k$ and $1 \leq j \leq l$;
(b) $V(G_i) \cap V(G_j) = \{s\}$ for $1 \leq i < j \leq k$.

Denote the working time of a sensor node by a unit time. In the ith unit time, the set of sensor nodes corresponding to the connected subgraph G_i are activated. These k disjoint node sets are activated successively. Therefore if $k = 2$ and the network graph G is $(l + \max\{1, l-4\})$-connected, or $k \geq 3$ and G is $(l(k-1)+1)$-connected, the lifetime of the WSN will be improved by k times. In fact, the preceding theorem implies more strong conclusion in wireless sensor networks, one is the targets can be arbitrary, the other is the network lifetime is maximized. For the case of $k = 2$, we conjecture that the connectivity $l+1$ is enough.

Conjecture 2 *Let G be a graph, $A_1, \ldots, A_l$ be any l pairwise disjoint vertex subsets with $|A_i| = 2$ for $1 \leq i \leq l$, $s \in V(G) \setminus \cup_{i=1}^{l} A_i$. If G is $(l+1)$-connected, then there exist 2 connected subgraphs G_1 and G_2 such that*

(*a*) $|V(G_i) \cap A_j| = 1$ *for* $i = 1, 2$ *and* $1 \le j \le l$;
(*b*) $V(G_1) \cap V(G_2) = \{s\}$.

For each l with $1 \le l \le 5$, Conjecture 2 is true since it is contained in Theorem 1. To conclude this section, we will show the sharpness of the condition "$(l(k-1)+1)$-connected" in Theorem 1. For each integer l and k, construct the graph $G(l,k)$ from $l(k-1)$-ary tree.

A *f-ary tree* is a rooted tree in which each vertex has no more than f children. A *full f-ary tree* is a f-ary tree where each vertex has either 0 or f children.

Let T be a full $l(k-1)$-ary tree of depth $l(k-1)-1$, and only the vertices in the $(l(k-1)-1)$th-layer have no children. Let T^* be a graph obtained from T which satisfies the following properties:

(i) the vertices in the dth-layer of T is joined to form a path $P_d = v_1^d \cdots v_{l^d(k-1)^d}^d$ for $1 \le d \le l(k-1)-1$;
(ii) let $v_{c_1}^d$ (resp. $v_{c_2}^d$) is a child of $v_{f_1}^{d-1}$ (resp. $v_{f_2}^{d-1}$) with $2 \le d \le l(k-1)-1$; if $f_1 < f_2$, then $c_1 < c_2$.

For $1 \le d \le l(k-1)-1$, call P_d as the dth-layer path of T^*, v_1^d the origin of P_d, and $v_{l^d(k-1)^d}^d$ the terminus of P_d.

Let $T_1^*, \ldots, T_l^*$ be l copies of T^*. For the dth-layer path of T_i^* with $1 \le d \le l(k-1)-1$ and $2 \le i \le l-1$, join the origin and the terminus of it to the terminus of dth-layer path of T_{i-1}^* and the origin of dth-layer path of T_{i+1}^*, respectively. Add $l(k-1)-1$ new vertices $s, u_0, u_1, \ldots, u_{l(k-1)-3}$ (if $l(k-1) \le 3$, we only add two vertices s, u_0), join s to the origin of the dth-layer path of T_1 for each d with $1 \le d \le l(k-1)-1$; join u_0 to s and all the vertices in the $(l(k-1)-1)$th-layer path of T_i^* for each i with $1 \le i \le l$; and for $1 \le j \le l(k-1)-3$, join u_j to the neighbors of u_0 except s. Denote the obtained graph by $G(l,k)$. The graph $G(1,4)$ is given in Fig. 1.

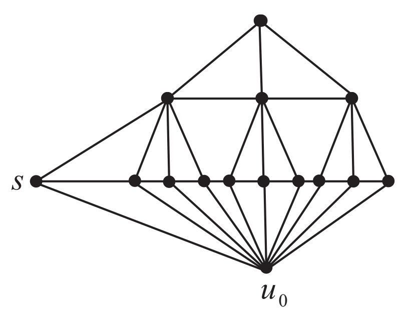

Fig. 1. The graph of $G(1,4)$.

For $1 \le i \le l$, denote the rooted vertex of T_i^* by v_i. In $G(l,k)$, denote the terminus of the dth-depth path of T_l^* by v_{l+d} for $1 \le d \le l(k-1)-1$, and let $u_0 = v_{lk}$. For $1 \le i \le l$, let $A_i = \{v_{(i-1)k+1}, v_{(i-1)k+2}, \ldots, v_{ik}\}$. $G(l,k)$ is

$l(k-1)$-connected, but we cannot find k connected subgraphs of $G(l,k)$ which satisfy (a) and (b) of Theorem 1. In fact, by the characteristics of $G(l,k)$, it needs at least $l(k-1)+1$ edges of $N_{G(l,k)}(s)$ to construct k connected subgraphs, but $|N_{G(l,k)}(s)| = l(k-1)$.

The rest of the paper is structured as follows. In section 2, we discuss previous work done in the coverage and connectivity in WSNs. In the following section, we give the proof of Theorem 1. The algorithms according to Theorem 1 are presented in section 4.

2 Related Work

Recently, a lot of research has been done to address the coverage problem in WSN. Related work has been done to implement energy efficient area coverage, connectivity or both (3; 5; 12; 14; 18). In some applications, when the network is sufficiently dense, area coverage can be approximated by guaranteeing point coverage. In this case, all the points of wireless devices could be used to represent the whole area, and the working sensors are supposed to cover all the sensors (13; 17).

Point coverage has also been considered. Cardei and Du (2) addressed the point coverage problem in which a set of targets with known locations needed to be monitored. They achieved energy efficiency by organizing the sensor nodes into a maximal number of disjoint set covers that are activated successively.

Design the set of active sensors as a connected dominating set (CDS) can assure coverage and connectivity. A distributed and localized protocol for constructing the CDS was proposed by Wu and Li (16). Dai and Wu (9) gave the dominating set algorithm to achieve connected point coverage. Wu *et al.* (15) also discuss the energy efficient dominating set coverage approach.

k-connected k-point coverage was also discussed. Zhou *et al.* (19) presented various algorithms to guarantee connected k-point coverage. Dai and Wu (10) have proposed several local algorithms to construct k-connected k-dominating set. Yang *et al.* (17) have proposed algorithms and solutions for k-(Connected) Coverage Set (k-CS/k-CCS) problems.

In many applications, the sensor nodes are dispersed very closely to each target and some other sensor nodes dispersed for transmitting information, which implies each target is in the sensing ranges of a lot of nodes. In this paper, we mainly consider these point coverage models, that is, each target with known location is monitored by lots of nodes in wireless sensor networks, we assume k nodes, and there are some nodes only for connection. We still organize the sensor nodes into disjoint sets to achieve point coverage and connectivity.

3 Proof of Theorem 1

Let T be a tree, $v_0v_1\cdots v_n$ a path in T. Denote the sub-path $v_iv_{i+1}\cdots v_j$ by $T[v_i,v_j]$, $v_{i+1}\cdots v_j$ by $T(v_i,v_j]$, and $v_i\cdots v_{j-1}$ by $T[v_i,v_j)$. When a path P is *internally vertex disjoint* to a graph G, it is simplified as P is IVD to G; when a sequence of paths $P_1,\ldots,P_n$ is *pairwise internally vertex disjoint*, it is simplified as $P_1,\ldots,P_n$ is PIVD. If a vertex v is in a subgraph G', we still call that v can be connected to G'.

Let $A_i=\{v_1^i,v_2^i,\ldots,v_k^i\}$ for $1\le i\le l$.

Case 1. $k=2$ and G is $(l+\max\{1,l-4\})$-connected.

If $l=1$, it is clear.

If $l=2$, then G is 3-connected. By the connectivity, there exist 3 PIVD paths P_1^1,P_1^2 and P_2^2 connecting s to v_1^1,v_1^2 and v_2^2, respectively. And we also may assume v_2^1 isn't in P_1^1, and it can be connected to P_1^2 or P_2^2 by some path Q, say P_1^2, which is IVD to P_1^1,P_1^2 and P_2^2. And then $G_1=P_1^1\cup P_2^2, G_2=Q\cup P_1^2$ are the two connected subgraphs satisfying (a) and (b).

If $l=3$, then G is 4-connected. By the connectivity, there exist 4 PIVD paths P_1^1,P_1^2,P_1^3 and P_2^3 connecting s to v_1^1,v_1^2,v_1^3 and v_2^3, respectively. And we may assume that v_2^i isn't in P_1^i for $i=1,2$. Suppose there exist two paths Q_1 and Q_2, which are IVD to P_1^1,P_1^2,P_1^3 and P_2^3, connecting v_2^1 and v_2^2 to the same path P_1^3 or P_2^3, respectively, say P_1^3. Then $G_1=Q_1\cup Q_2\cup P_1^3, G_2=P_1^1\cup P_1^2\cup P_2^3$ are the two connected subgraphs satisfying (a) and (b).

Now suppose v_2^1 and v_2^2 are connected to the different paths by the paths IVD to P_1^1,P_1^2,P_1^3 and P_2^3. Assume v_2^1 is connected to P_1^3 by Q_1, and v_2^2 is connected to P_1^1 by Q_2. Then $G_1=P_1^1\cup Q_2\cup P_2^3, G_2=Q_1\cup P_1^2\cup P_1^3$ are the two connected subgraphs satisfying (a) and (b). Similarly, for other cases, we can find two connected subgraphs satisfying (a) and (b).

If $l=4$, then G is 5-connected. And then there exist 5 PIVD paths P_1^1,P_1^2,P_1^3, P_1^4,P_2^4 connecting s to $v_1^1,v_1^2,v_1^3,v_1^4,v_2^4$, respectively. And for $i=1,2,3$, we may assume that v_2^i isn't in P_1^i.

If one vertex of $\{v_2^1,v_2^2,v_2^3\}$ can be connected to P_1^4 or P_2^4 by a path which is IVD to P_1^1,P_1^2,P_1^3,P_1^4 and P_2^4, assume v_2^1 is connected to P_1^4 by Q_1, then let $P_1^*=Q_1\cup P_1^4$ and $P_2^*=P_1^1\cup P_2^4$. And then, similar to the case of $l=3$, we can find the two connected subgraphs satisfying (a) and (b).

Now suppose no vertex of $\{v_2^1, v_2^2, v_2^3\}$ can be connected to P_1^4 or P_2^4 by the path which is IVD to $P_1^1, P_1^2, P_1^3, P_1^4$ and P_2^4.

Assume both of two vertices of $\{v_2^1, v_2^2, v_2^3\}$ can be connected to one same path of $\{P_1^1, P_1^2, P_1^3\}$ by the paths which are IVD to $P_1^1, P_1^2, P_1^3, P_1^4$ and P_2^4, say v_2^1 and v_2^2 are connected to P_1^3 by Q_1 and Q_2, respectively. And if v_2^3 is connected to P_1^1 by Q_3 IVD to $P_1^1, P_1^2, P_1^3, P_1^4, P_2^4, Q_1$ and Q_2. Then $G_1 = Q_1 \cup Q_2 \cup P_1^3 \cup P_1^4$ and $G_2 = P_1^1 \cup P_1^2 \cup Q_3 \cup P_2^4$ are the two connected subgraphs satisfying (a) and (b). And for other cases, similarly, we can find two connected subgraphs satisfying (a) and (b).

Otherwise, we may assume v_2^i can only be connected to P_1^{i+1} by the paths IVD to $P_1^1, P_1^2, P_1^3, P_1^4, P_2^4$, where $i = 1, 2, 3$ and $i + 1$ is taken module 3. Choose a path Q_i such that v_2^i is connected to P_1^{i+1} by it, and Q_i is IVD to $P_1^1, P_1^2, P_1^3, P_1^4$ and P_2^4. Q_1, Q_2 and Q_3 are PIVD.

For $1 \leq i \leq 3$, let s_1^i be the vertex of P_1^i such that for any vertex $v \in V(P_1^i)$ if v can be connected to $\{v_1^1, v_2^1, v_1^2, v_2^2, v_1^3, v_2^3\}$ by a path which is IVD to $P_1^1, P_1^2, P_1^3, P_1^4, P_2^4, Q_1, Q_2, Q_3$, then $d_{P_1^i}(s_1^i, s) \leq d_{P_1^i}(v, s)$.

Then, for $i = 1, 2, 3$, $s_1^i \neq s$. Suppose a vertex of $P_1^1(s_1^1, v_1^1]$ can be connected to P_1^4 by the path Q. Assume v_2^3 can be connected to s_1^1 by the path Q' which is IVD to $P_1^1, P_1^2, P_1^3, P_1^4, P_2^4, Q_1, Q_2, Q_3, Q$. Then let $G_1 = (P_1^1 - P_1^1[s, s_1^1]) \cup Q \cup Q_2 \cup P_1^3 \cup P_1^4$ and $G_2 = Q_1 \cup P_1^2 \cup Q' \cup P_1^1[s, s_1^1] \cup P_2^4$. Similarly, for other cases we can easily find G_1 and G_2 which are the connected subgraphs satisfying (a) and (b).

If $l \geq 5$, then G is $(2l-4)$-connected. There exist $2l-4$ PIVD paths $P_1^1, \ldots, P_1^4$, $P_1^5, P_2^5, \ldots, P_1^l, P_2^l$ connecting s to $v_1^1, \ldots, v_1^4, v_1^5, v_2^5, \ldots, v_1^l, v_2^l$, respectively. And for $i = 1, \ldots, 4$, we may assume that v_2^i isn't in P_1^i.

If one vertex of $\{v_2^1, v_2^2, v_2^3, v_2^4\}$ can be connected to one path of $\{P_1^5, P_2^5, \ldots, P_1^l, P_2^l\}$, say v_2^1 is connected to P_1^5 by Q_1, which is IVD to $P_1^1, \ldots, P_1^4, P_1^5, P_2^5, \ldots, P_1^l, P_2^l$, then let $P_1^* = P_1^5 \cup \cdots \cup P_1^l \cup Q_1$ and $P_2^* = P_2^5 \cup \cdots \cup P_2^l \cup P_1^1$. Then similar to the case of $l = 4$, we can find two connected subgraphs satisfying (a) and (b).

For other cases (shown in Fig. 2), similar to the previous cases, there exist two connected subgraphs satisfying (a) and (b).

Case 2. $k \geq 3$ and G is $(l(k-1)+1)$-connected.

By the connectivity of G, there exist $l(k-1)+1$ PIVD paths connecting s to each vertex of $\{v_1^1, v_2^1, \ldots, v_{k-1}^1, \ldots, v_1^{l-1}, v_2^{l-1}, \ldots, v_{k-1}^{l-1}, v_1^l, v_2^l, \ldots, v_k^l\}$. For $1 \leq i \leq l-1$ and $1 \leq j \leq k-1$, or $i = l$ and $1 \leq j \leq k$, let P_j^i be the (s, v_j^i)-path. For $i \neq l$, we may assume v_k^i isn't in P_j^i, and v_k^i should be connected to

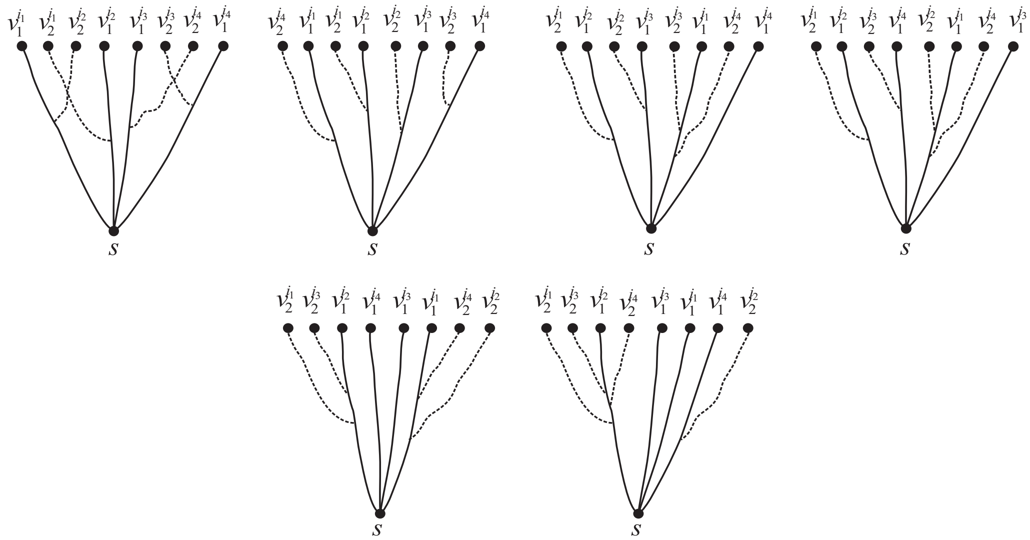

Fig. 2. A few cases of $l \geq 5$.

one such path since G is connected.

Claim 1. For $1 \leq i \leq l-1$, we can find connected subgraphs $G_1^i, G_2^i, \ldots, G_{i^*}^i, \ldots, G_k^i$ such that G_j^i contains s and only one vertex v_j^i of A_i for $1 \leq j \leq k$; in particular, $G_{i^*}^i$ also contains some vertices of $\cup_{i \neq j} A_j$.

Proof. We show it by giving an algorithm.

(1) Set $a_0 = 1$, $I = \emptyset$.

(2) For $1 \leq i \leq l$, $1 \leq j \leq k$, set $G_j^i = P_j^i$; especially, if P_k^i doesn't exist, set G_k^i and P_k^i be the single vertex v_k^i. Set $\mathcal{G} = \{P_1^1, \ldots, P_k^1, \ldots, P_1^l, \ldots, P_k^l\}$.

(3) If $P_k^{a_0}$ contains the vertex s or one vertex of A_l, replace I by $I \cup \{a_0\}$ and set $a_0^* = k$. Go to step (9).

(4) Set $m = 1$, $p_m = 0$, $h_{p_m}^m = 1$, and $\mathcal{G}_{a_0} = \{P_{(0,h_{p_m}^m,m)}^{a_0}\} = \{P_{(0,1,1)}^{a_0}\} = \{P_k^{a_0}\}$.

(5) Replace m by $m+1$. Assume $P_k^{a_0}$ contains the vertex $v_k^{a_0}$ and some vertices of A_i for $i = a_1, \ldots, a_{p_m}$, $p_m \leq l$.

If $P_k^{a_0}$ can be connected to some subgraph P_t^r by the path Q with $r \neq a_0, a_1, \ldots, a_{p_m}$, where Q is IVD to any graph of $\mathcal{G}$, then replace $P_k^{a_0}$ by $P_k^{a_0} \cup Q \cup P_t^r$, P_t^r by $P_k^{a_0}$, I by $I \cup \{a_0\}$ and set $a_0^* = k$. Go to step (9).

(6) Otherwise, choose all the subgraphs $P_{(0,1,m+1)}^{a_0}, \ldots, P_{(0,h_0^{m+1},m+1)}^{a_0}, P_{(1,1,m+1)}^{a_1}, \ldots, P_{(1,h_1^{m+1},m+1)}^{a_1}, \ldots, P_{(p_{m+1},1,m+1)}^{a_{p_{m+1}}}, \ldots, P_{(p_{m+1},h_{p_{m+1}}^{m+1},m+1)}^{a_{p_{m+1}}}$ such that at least one graph of $\mathcal{G}_{a_0}$ can be connected to them by some paths, which are IVD to any graph of $\mathcal{G}$.

For $0 \leq i \leq p_{m+1}$ and $1 \leq j \leq h_i^{m+1}$, let $s_{(i,j,m+1)}$ be the vertex in $P_{(i,j,m+1)}^{a_i}$ such that if $v \in V(P_{(i,j,m+1)}^{a_i})$ can be connected to one graph of $\mathcal{G}_{a_0}$, then $d_{P_{(i,j,m+1)}^{a_i}}(s_{(i,j,m+1)}, s) \leq d_{P_{(i,j,m+1)}^{a_i}}(v, s)$.

For $1 \leq i' \leq p_m$, $1 \leq j' \leq h_{i'}^m$, $1 \leq i \leq p_{m+1}$, $1 \leq j \leq h_i^{m+1}$, choose the path $Q^{(i',j',m)}_{(i,j,m+1)}$ (if exists), which is IVD to any graph of $\mathcal{G}$, such that $P^{a_{i'}}_{(i',j',m)}$ can be connected to $s_{(i,j,m+1)}$ by it; and replace $\mathcal{G}$ by $\mathcal{G}\cup\{Q^{(i',j',m)}_{(i,j,m+1)}\}$; set $P^{(i',j',m)}_{(i,j,m+1)} = P^{a_{i'}}_{(i',j',m)} \cup Q^{(i',j',m)}_{(i,j,m+1)} \cup P^{a_i}_{(i,j,m+1)}[s, s_{(i,j,m+1)}]$, replace $P^{a_i}_{(i,j,m+1)}$ by $P^{a_i}_{(i,j,m+1)} - P^{a_i}_{(i,j,m+1)}[s, s_{(i,j,m+1)}]$. Replace $\mathcal{G}_{a_0}$ by $\{P^{a_0}_{(0,1,m+1)}, \ldots, P^{a_0}_{(0,h_0^{m+1},m+1)}, P^{a_1}_{(1,1,m+1)}, \ldots, P^{a_1}_{(1,h_1^{m+1},m+1)}, \ldots, P^{a_{p_{m+1}}}_{(p_{m+1},1,m+1)}, \ldots, P^{a_{p_{m+1}}}_{(p_{m+1},h^{m+1}_{p_{m+1}},m+1)}\}$, and m by $m+1$.

(7) If there exists $P^{a_{i^*}}_{(i^*,j^*,m)}$ with $1 \le i^* \le p_m$ and $1 \le j^* \le h_i^m$ such that it can be connected to P_t^r by the path Q which is IVD to any graph of $\mathcal{G}$, where P_t^r satisfies that there doesn't exist A_x with $1 \le x \le l$ such that both $|A_x \cap V(P^{a_{i^*}}_{(i^*,j^*,m)})| \ge 1$ and $|A_x \cap V(P_t^r)| \ge 1$, then replace $P^{a_{i^*}}_{(i^*,j^*,m)}$ by $P^{a_{i^*}}_{(i^*,j^*,m)} \cup Q \cup P_t^r$, P_t^r by $P^{a_{i^*}}_{(i^*,j^*,m)}$.

Choose a sequence $i_1 = 0, \ldots, i_m = i^*, j_1 = 1, \ldots, j_m = j^*$ (some of them may be equal) such that for $1 \le x \le m-1$ the path $Q^{(i_x,j_x,x)}_{(i_{x+1},j_{x+1},x+1)}$ exists.

Then for $1 \le x \le m$, replace P_j^i by G_j^i for $i \ne a_{i_x}$ and $j \ne (i_x, j_x, x)$; for $1 \le x \le m-1$, replace $P^{a_{i_x}}_{(i_x,j_x,x)}$ by $P^{(i_x,j_x,x)}_{(i_{x+1},j_{x+1},x+1)}$; if $P^{a_{i_x}}_{(i_x,j_x,x)}$ contains the vertex $v_j^i \in A_i$ with $i \ne a_{i_x}$, replace P_j^i by $P^{a_{i_x}}_{(i_x,j_x,x)}$. Set $a^*_{i_m} = (i_m, j_m, m)$, replace I by $I \cup \{a_{i_m}\}$, and go to step (9).

(8) Otherwise, go to step (6).

(9) If $I = \{1, 2, \ldots, l-1\}$, stop.

(10) Otherwise, choose an integer from $\{1, 2, \ldots, l-1\} \setminus I$, replace a_0 by it, go to step (2).

By the preceding algorithm, for each i with $1 \le i \le l$, j with $1 \le j \le k$, we can find subgraph G_j^i containing s and only one vertex v_j^i of A_i; in particular, for $1 \le i \le l-1$ there exist $r \ne i$ and $t \ne r^*$ such that $G^i_{i^*} = G^r_t$.□

Let $A = \{v^1_{1^*}, v^2_{2^*}, \ldots, v^{l-1}_{(l-1)^*}\}$. For $1 \le j \le k$, let $G_j^{(0)} = G_j^l$.

Suppose we have found the connected subgraphs $G_1^{(r)}, \ldots, G_k^{(r)}$ satisfying

(i) $V(G^{(r)}_{j_1}) \cap V(G^{(r)}_{j_2}) = \{s\}$ for $1 \le j_1 < j_2 \le k$;

(ii) for any A_i with $1 \le i \le l$, and each j with $1 \le j \le k$, $G_j^{(r)}$ contains at most one vertex of A_i, and either all the vertices of $A_i \setminus \{v^i_{i^*}\}$ are in $\cup_{j=1}^k V(G_j^{(r)})$ or no vertex of $A_i \setminus \{v^i_{i^*}\}$ is in $\cup_{j=1}^k V(G_j^{(r)})$;

(iii) at most one vertex $v^i_{i^*}$ of A but not in $\cup_{j=1}^k V(G_j^{(r)})$ has the same superscript i to some vertices of $G_1^{(r)}, \ldots, G_{k-1}^{(r)}$ or $G_k^{(r)}$.

Suppose no vertex of $A \setminus \cup_{j=1}^k V(G_j^{(r)})$ has the same superscript to any vertex of $G_1^{(r)}, \ldots, G_{k-1}^{(r)}$ or $G_k^{(r)}$. If $\cup_{j=1}^l A_j \subseteq \cup_{j=1}^k V(G_j^{(r)})$, then $G_1^{(r)}, \ldots, G_k^{(r)}$ are the subgraphs satisfying (a) and (b). Otherwise, choose a vertex $v^i_{i'}$ not in $\cup_{j=1}^k V(G_j^{(r)})$ with $i' \ne i^*$. For $1 \le j \le k-1$, assume $G_j^{(r)}$ doesn't contain the vertex $v^i_{i^*}$. For $1 \le j < i^*$, let $G_j^{(r+1)} = G_j^{(r)} \cup G^i_j$; for $i^* \le j \le k-1$, let $G_j^{(r+1)} = G_j^{(r)} \cup G^i_{j+1}$. Then all the vertices of $A_i \setminus \{v^i_{i^*}\}$ are in $\cup_{j=1}^k V(G_j^{(r)})$.

Suppose $v^i_{i^*} \in A \setminus \cup_{j=1}^k V(G_j^{(r)})$ is of the same superscript i to some vertices of $G_1^{(r)}, \ldots, G_{k-1}^{(r)}$ or $G_k^{(r)}$. Assume $G^i_{i^*} = G^t_{t'}$, where $G^t_{t'}$ is a subgraph of $G - (\cup_{j=1}^k G_j^{(r)} - s)$ and $t' \ne t^*$. By the property (ii), assume $G_j^{(r)}$ contains the vertex $v^i_{i_j}$ with $i_j \ne i^*$ for $1 \le j \le k-1$.

If $v^t_{t^*}$ isn't in any graph $G_j^{(r)}$ with $1 \le j \le k$, let $G_k^{(r+1)} = G_k^{(r)} \cup G^t_{t'}$. Without loss of generality, assume $t^* > t'$. For $1 \le j < t'$, let $G_j^{(r+1)} = G_j^{(r)} \cup G^t_j$; for $t' \le j < t^*$, let $G_j^{(r+1)} = G_j^{(r)} \cup G^t_{j+1}$; for $t^* \le j \le k-2$, let $G_j^{(r+1)} = G_j^{(r)} \cup G^t_{j+2}$. Then all the vertices of A_i and $A_t \setminus \{v^t_{t^*}\}$ are in $\cup_{j=1}^k V(G_j^{(r)})$.

Otherwise, assume there exists m_1 with $1 \le m_1 \le k$ such that $G^t_{t^*} = G^s_{s'}$, and $G^s_{s'}$ is a subgraph of $G^{(r)}_{m_1}$. If $m_1 \ne k$, then $G_k^{(r)}$ doesn't contain the vertices $v^t_{t^*}$ and v^i_j for $1 \le j \le k$ and $j \ne i^*$. Then let $G_k^{(r+1)} = G_k^{(r)} \cup G^t_{t'}$. And let $\{G^{(r)}_{r_1}, \ldots, G^{(r)}_{r_{k-2}}\} = \{G_1^{(r)}, G_2^{(r)}, \ldots, G_k^{(r)}\} \setminus \{G^{(r)}_{m_1}, G^{(r)}_k\}$. For $1 \le j < t'$, let $G^{(r+1)}_{r_j} = G^{(r)}_{r_j} \cup G^t_j$; for $t' \le j < t^*$, let $G^{(r+1)}_{r_j} = G^{(r)}_{r_j} \cup G^t_{j+1}$; for $t^* \le j \le k-2$, let $G^{(r+1)}_{r_j} = G^{(r)}_{r_j} \cup G^t_{j+2}$. Then all the vertices of A_i and A_t are in $\cup_{j=1}^k V(G_j^{(r)})$.

Otherwise, $G^{(r)}_{m_1} = G^{(r)}_k$, which contains the vertex $v^t_{t^*}$ but no vertex with superscript i, and for $1 \le j \le k-1$, $G_j^{(r)}$ contains the vertex $v^i_{i_j}$ with $i_j \ne i^*$. Since $k \ge 3$, we can find a subgraph $G^{(r)}_{m_2}$ such that $G^s_{s''}$ is a subgraph of $G^{(r)}_{m_2}$, and $G^s_{s''}$ doesn't contain any vertex of superscript i or t. Let $G^{(r+1)_1}_{m_2} = (G^{(r)}_{m_2} - G^s_{s''}) \cup G^s_{s'}$, $G^{(r+1)_1}_{m_1} = (G^{(r)}_{m_1} - G^s_{s'}) \cup G^s_{s''}$. If $G^{(r+1)_1}_{m_1}$ contains the vertices $v^{a_1}_{a'_1}, v^{a_1}_{a^*_1}, \ldots, v^{a_n}_{a'_n}, v^{a_n}_{a^*_n}$ with $1 \le n \le l$, then let $G^{(r+1)_2}_{m_1} = G^{(r+1)_1}_{m_1} - (\cup_{j=1}^n G^{a_j}_{a'_j} - s)$, and $G^{(r+1)_2}_{m_2} = G^{(r+1)_1}_{m_2} \cup (\cup_{j=1}^n G^{a_j}_{a'_j})$. And then consider it similarly for $G^{(r+1)_2}_{m_2}$, we have $G^{(r+1)_3}_{m_1}$ and $G^{(r+1)_3}_{m_2}$. Do it repeatedly. By the characteristics of $G^{(r)}_{m_1}$ and $G^{(r)}_{m_2}$, it will terminate with some integer $b \ge 1$. Let $G^{(r+1)}_{m_1} = G^{(r+1)_b}_{m_1} \cup G^t_{t'}$, $G^{(r+1)}_{m_2} = G^{(r+1)_b}_{m_2}$. And let $\{G^{(r)}_{r_1}, \ldots, G^{(r)}_{i_{k-2}}\} = \{G_1^{(r)}, G_2^{(r)}, \ldots, G_k^{(r)}\} \setminus \{G^{(r)}_{m_1}, G^{(r)}_{m_2}\}$. For $1 \le j < t'$, let $G^{(r+1)}_{r_j} = G^{(r)}_{r_j} \cup G^t_j$; for $t' \le j < t^*$, let $G^{(r+1)}_{r_j} = G^{(r)}_{r_j} \cup G^t_{j+1}$; for $t^* \le j \le k-2$, let $G^{(r+1)}_{r_j} = G^{(r)}_{r_j} \cup G^t_{j+2}$.

Since G is a limited graph, we can find an integer p such that $G_1^{(p)}, \ldots, G_k^{(p)}$ which are the connected subgraphs satisfying (a) and (b).

4 The Related Algorithms

According to the proof of Theorem 1, first we give the algorithm of the case of $k = 2$ and G is $(l + \max\{1, l-4\})$-connected.

Algorithm 1.

(1) Set $a = \max\{1, l-4\}$.
(2) Choose $l+a$ PIVD paths $P_1^1, \ldots, P_1^l, P_2^{l-a+1}, \ldots, P_2^l$ connecting s to $v_1^1, \ldots, v_1^l, v_2^{l-a+1}, \ldots, v_2^l$ such that for $1 \le i \le l-a$, v_2^i isn't in P_1^i and it can be connected to one of such path.
(3) For $1 \le i \le l-a$, choose a path Q_i which is IVD to $P_1^1, \ldots, P_1^l, P_2^{l-a+1}, \ldots, P_2^l$, such that v_2^i is connected to P_h^j by Q_i with $i \neq j$ and $h = 1, 2$. Replace P_h^j by $P_h^j \cup Q_i$, and set $P_2^i = P_h^j$.
(4) Set $G_1 = \cup_{i=l-a+1}^{l} P_1^i$, $G_2 = \cup_{i=l-a+1}^{l} P_2^i$.
(5) For $1 \le i \le l-a$, $j = 1, 2$, if $v_2^i \in V(G_j)$, and there doesn't exist A_x with $1 \le x \le l-a$ such that both $|V(P_1^i) \cap A_x| = 1$ and $|V(G_{j+1}) \cap A_x| = 1$, then replace G_{j+1} by $G_{j+1} \cup P_1^i$, where $j+1$ is taken module 2.
(6) If $|A_i \cap V(G_j)| = 1$ for $1 \le i \le l-a$ and $j = 1, 2$, stop.
(7) If there exist $v_2^i \in \{v_2^1, \ldots, v_2^{l-a}\} \setminus V(G_1 \cup G_2)$ and $j \in \{1, 2\}$ such that $v_1^h \notin V(G_j)$, $P_1^h = P_2^i$, and there doesn't exist A_x with $1 \le x \le l-a$ such that both $|V(P_2^i) \cap A_x| = 1$ and $|V(G_j) \cap A_x| = 1$, then replace G_j by $G_j \cup P_2^i$. Go to step (5).
(8) Choose an integer $x_0 \in \{1, 2, 3, 4\}$ such that no vertex of $\{v_2^1, v_2^2, v_2^3, v_2^4\}$ is connected to $P_1^{x_0}$ by Q_1, Q_2, Q_3 or Q_4.
(9) Set $\{x_1, x_2, x_3\} = \{1, 2, 3, 4\} \setminus \{x_0\}$. For $i = x_1, x_2, x_3$, set s_1^i be the vertex of P_1^i such that for any vertex $v \in V(P_1^i)$ if v can be connected to $\{v_1^{x_1}, v_2^{x_1}, v_1^{x_2}, v_2^{x_2}, v_1^{x_3}, v_2^{x_3}\}$ by a path which is IVD to $P_1^1, P_2^1, \ldots, P_1^l, P_2^l$, then $d_{P_1^i}(s_1^i, s) \le d_{P_1^i}(v, s)$.
(10) If $s_1^{x_1} = s_1^{x_2} = s_1^{x_3} = s$, choose a vertex $v_{i'}^i \in \{v_1^{x_1}, v_2^{x_1}, v_1^{x_2}, v_2^{x_2}, v_1^{x_3}, v_2^{x_3}\}$ and the path Q which is IVD to $P_1^1, P_2^1, \ldots, P_1^l, P_2^l$ such that $v_{i'}^i$ is connected to s by Q. For the subgraph $P_{j'}^j$ with $P_{i'}^i = P_{j'}^j$, replace $P_{j'}^j$ by $P_{j'}^j - v_{i'}^i$, and replace $P_{i'}^i$ by Q. Go to step (4).
(11) Otherwise, choose $P_{r'}^r$ with $1 \le r \le l$ and $r \neq x_1, x_2, x_3$, $r' \in \{1, 2\}$, a vertex v of $P_1^x - P_1^x[s, s_1^x]$ with $x \in \{x_1, x_2, x_3\}$, and the path Q which is IVD to $P_1^1, P_2^1, \ldots, P_1^l, P_2^l$ such that v is connected to $P_{r'}^r$ by Q. Choose a vertex $v_{i'}^i \in \{v_1^{x_1}, v_2^{x_1}, v_1^{x_2}, v_2^{x_2}, v_1^{x_3}, v_2^{x_3}\}$ and the path Q' which is IVD to $P_1^1, P_2^1, \ldots, P_1^l, P_2^l, Q$ such that $v_{i'}^i$ is connected to s_1^x by Q'.
(12) If $v_{i'}^i \in V(P_1^x)$ and $v_{i'}^i = v$, then for the subgraph $P_{t'}^t$ with $P_{t'}^t = P_{i'}^i$ and $i \neq t$, replace $P_{i'}^i$ by $Q \cup P_{r'}^r$, $P_{t'}^t$ by $P_{t'}^t - v_{i'}^i$; for any subgraph $P_{j'}^j$ with $P_{j'}^j = P_{r'}^r$, replace $P_{r'}^r$ and $P_{j'}^j$ by $P_{i'}^i$. Go to step (4).
(13) If $v_{i'}^i \in V(P_1^x)$ and $v_{i'}^i \neq v$, then for the subgraph $P_{t'}^t$ with $P_{t'}^t = P_{i'}^i$ and $i \neq t$, replace $P_{i'}^i$ by $Q' \cup P_{i'}^i[s, s_1^x]$, $P_{t'}^t$ by $(P_{t'}^t - P_{i'}^i) \cup Q \cup P_{r'}^r$; for any subgraph $P_{j'}^j$ with $P_{j'}^j = P_{r'}^r$, replace $P_{r'}^r$ and $P_{j'}^j$ by $P_{t'}^t$. Go to step (4).

(14) If $v^i_{i'} \notin V(P^x_1)$, then for the subgraph $P^t_{t'}$ with $P^t_{t'} = P^i_{i'}$ and $i \neq t$, replace $P^i_{i'}$ by $Q' \cup P^x_1[s, s^x_1]$, $P^t_{t'}$ by $P^t_{t'} - v^i_{i'}$. And for the subgraph $P^j_{j'}$ with $P^j_{j'} = P^x_1$ and $j \neq x$ or $P^j_{j'} = P^r_{r'}$, replace P^x_1 by $(P^x_1 - P^i_{i'}) \cup Q \cup P^r_{r'}$, replace $P^r_{r'}$ and $P^j_{j'}$ by P^x_1. Go to step (4). $\square$

For simplicity, call the algorithm used in the proof of Claim 1 as Algorithm **A**. According to the proof of Theorem 1, the rest part of the algorithm of the case of $k \geq 3$ and G is $(l(k-1)+1)$-connected to find the disjoint connected subgraphs will be given below.

Algorithm 2.

(1) Find $l(k-1)+1$ PIVD paths $P^1_1, \ldots, P^1_{k-1}, \ldots, P^{l-1}_1, \ldots, P^{l-1}_{k-1}, P^l_1, P^l_2, \ldots, P^l_k$ such that P^i_j is a (s, v^i_j)-path for $1 \leq i \leq l-1$ and $1 \leq j \leq k-1$, or $i = l$ and $1 \leq j \leq k$, and v^i_k isn't in P^i_j for $i \neq l$.

(2) Using Algorithm **A**, find the subgraph G^i_j containing v^i_j and s for $1 \leq i \leq l$ and $1 \leq j \leq k$; and for $1 \leq i \leq l-1$ there exist $r \neq i$ and $t \neq r^*$ such that $G^i_{i^*}$ is equal to G^r_t.

(3) For $1 \leq i \leq k$, set $G_i = G^l_i$. Set $A = \{v^1_{1^*}, v^2_{2^*}, \ldots, v^{l-1}_{(l-1)^*}\}$.

(4) If $\cup^l_{i=1} A_i \subseteq \cup^k_{i=1} V(G_i)$, stop.

(5) If there is no vertex of $A \setminus \cup^k_{i=1} V(G_i)$ which has the same superscript to any vertex of $G_1, \ldots, G_{k-1}$ or G_k, choose a vertex $v^r_{r'}$ such that $v^r_{r'} \in \cup^l_{i=1} A_i \setminus \cup^k_{i=1} V(G_i)$ and $r' \neq r^*$; and choose $k-1$ subgraphs $G_{i_1}, \ldots, G_{i_{k-1}}$ such that for $1 \leq j \leq k-1$, G_{i_j} doesn't contain the vertex $v^r_{r^*}$. For $1 \leq j < r^*$, replace G_{i_j} by $G_{i_j} \cup G^r_j$; for $r^* \leq j \leq k-1$, replace G_{i_j} by $G_{i_j} \cup G^r_{j+1}$. Go to step (4).

(6) Otherwise, choose $v^r_{r^*} \in A \setminus \cup^k_{i=1} V(T_i)$ which is of the same superscript r to some vertices of $G_1, \ldots, G_{k-1}$ or G_k. Choose the subgraph $G^t_{t'}$ with $t' \neq t^*$ from $G - (\cup^k_{i=1} G_i - s)$ such that $G^r_{r^*} = G^t_{t'}$.

(7) If $v^t_{t^*}$ is in $G_1, \ldots, G_{k-1}$ or G_k, go to step (9). Otherwise, choose a subgraph $G_{i'}$ from $\{G_1, G_2, \ldots, G_k\}$ such that $G_{i'}$ doesn't contain the vertex $v^r_{r'}$ for each r' with $1 \leq r' \leq k$ and $r' \neq r^*$. Replace $G_{i'}$ by $G_{i'} \cup G^t_{t'}$. Choose $k-2$ subgraphs $G_{i_1}, \ldots, G_{i_{k-2}}$ such that for each j with $1 \leq j \leq k-2$, G_{i_j} doesn't contain any vertex with superscript t.

(8) Set $a = \min\{t', t^*\}$ and $b = \max\{t', t^*\}$. For $1 \leq j < a$, replace G_{i_j} by $G_{i_j} \cup G^t_j$; for $a \leq j < b$, replace G_{i_j} by $G_{i_j} \cup G^t_{j+1}$; for $b \leq j \leq k-2$, replace G_{i_j} by $G_{i_j} \cup G^t_{j+2}$. Go to step (4).

(9) Choose $G^s_{s'}$ and G_{m_1} such that $G^t_{t^*} = G^s_{s'}$ and $G^s_{s'}$ is a subgraph of G_{m_1}.

(10) If there exist a subgraph G_h with $h \neq m_1$ such that G_h doesn't contain the vertices $v^t_{t^*}$ and any vertex with superscript r, then replace G_h by $G_h \cup G^t_{t'}$. Set $\{G_{i_1}, \ldots, G_{i_{k-2}}\} = \{G_1, G_2, \ldots, G_k\} \setminus \{G_{m_1}, G_h\}$. Go to step (8).

(11) Otherwise, choose $G^s_{s''}$ and G_{m_2} such that $G^s_{s''}$ is a subgraph of G_{m_2}, and $G^s_{s''}$ doesn't contain any vertex with the superscript r or t. Replace G_{m_2} by $(G_{m_2} - G^s_{s''}) \cup G^s_{s'}$, and replace G_{m_1} by $(G_{m_1} - G^s_{s'}) \cup G^s_{s''}$.

(12) For $i = 1, 2$, if there doesn't exist A_j with $1 \le j \le l$ such that $|V(G_{m_i}) \cap A_j| = 2$, go to step (14).

(13) Otherwise, $V(G_{m_i})$ contains $v^{a_1}_{a'_1}, v^{a_1}_{a^*_1}, \ldots, v^{a_n}_{a'_n}, v^{a_n}_{a^*_n}$ with $1 \le n \le l$, then replace G_{m_i} by $G_{m_i} - (\cup^n_{j=1} G^{a_i}_{a'_i} - s)$, and replace $G_{m_{i+1}}$ by $G_{m_{i+1}} \cup (\cup^n_{j=1} G^{a_i}_{a'_i})$, where $i+1$ is taken module 2. Go to step (12).

(14) Replace G_{m_1} by $G_{m_1} \cup G^t_{t'}$, and set $\{G_{i_1}, \ldots, G_{i_{k-2}}\} = \{G_1, G_2, \ldots, G_k\} \setminus \{G_{m_1}, G_{m_2}\}$. Go to step (8).□

5 Acknowledgements

We would like to thank the referees for their valuable suggestions.

References

[1] J. A. Bondy, U. S. R. Murty, Graph Theory with Applications. Macmillan, New York, 1976.

[2] M. Cardei, D. Du, Improving Wireless Sensor Network Lifetime Through Power Aware Organazation, *ACM Wireless Network*, 2005, Vol. 11: pp. 333-340.

[3] M. Cardei, D. MacCallum, X. Cheng, M. Min, X. Jia, D. Li, D. Du, Wireless Sensor Networks with Energy Efficient Organization. *Journal of Interconnection Networks*, 2002, Vol. 3(3-4): pp. 213-229.

[4] M. Cardei, J. Wu, Energy-Efficient Coverage Problem in Wireless Ad-Hoc Sensor Networks. *Computer Communications*, 2006, Vol. 29: pp. 413-420.

[5] B. Chen, K. Jamieson, H. Balakrishnan, R. Morris, Span: An Energy-Efficient Coordination Algorithm for Topology Maintenance in Ad Hoc Wireless Networks. *ACM Wireless Networks Journal*, 2002, Vol. 8(5): pp. 481-494.

[6] I. Chlamtac, C. Petrioli, J. Redi, Energy-Conserving Access Protocols for Identification Networks. *IEEE/ACM Transactions on Networking*, 1999, Vol. 7, No. 1: pp. 51-59.

[7] D. Culer, D. Estrin, M. Srivastava, Overview of Sensor Networks. *IEEE Computer*, 2004, Vol. 8: pp. 41-49.

[8] F. Dai, J. Wu, Distributed Dominant Pruning in Ad Hoc Wireless Networks, *Proceedings of IEEE International Conference on Communications*, 2003, pp. 353-357.

[9] F. Dai, J. Wu, An Extended Localized Algorithm for Connected Dominating Set Formation in Ad Hoc Wireless Networks. *IEEE Transactions on Parallel and Distributed Systems*, 2004, Vol. 15: pp. 908-920.

[10] F. Dai, J. Wu, On Constructing k-Connected k-Dominating Set in Wireless Networks. *In IPDPS, IEEE Computer Society*, 2005.

[11] K. Römer, F. Mattern, The Design Space of Wireless Sensor Networks. *IEEE Wireless Communications*, 2004, Vol. 11: pp. 54-61.

[12] S. Slijepcevic, M. Potkonjak, Power Efficient Organization of Wireless Sensor Networks. *Proceedings of IEEE International Conference on Communications 2*, 2001, pp. 472-476.

[13] D. Tian, N. Georganas, A Coverage-Preserving Node Scheduling Scheme for Large Wireless Sensor networks. *In Proceedings of the 1st ACM Workshop on Wireless Sensor Networks and Applications*, 2002, pp. 32-41.

[14] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, C. Gill, Integrated Coverage and Connectivity Configuration in Wireless Sensor Networks. *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, 2003, pp. 28-39.

[15] J. Wu, F. Dai, M. Gao, I. Stojmenovic, On Calculating Power-Aware Connected Dominating Sets for Efficient Routing in Ad Hoc Wireless Networks. *Journal of Communications and Networks*, 2002, Vol. 4 PP. 1-12.

[16] J. Wu, H. Li, On Calculating Connected Dominating Set for Efficient Routing in Ad Hoc Wireless Networks. *Proceedings of the Third International Workshop on Discrete Algorithms and Mathods for Mobile Computing and Communications*, 1999, pp. 7-14.

[17] S. Yang, F. Dai, M. Cardei, J. Wu, F. Patterson, On Connected Multiple Point Coverage in Wireless Sensor Networks. *Journal of Wireless Information Networks*, 2006, Vol. 13: pp. 289-301.

[18] H. Zhang, J. Hou, Maintaining Sensing Coverage and Connectivity in Large Sensor Networks. *Ad Hoc and Sensor Wireless Networks*, 2005, Vol. 1: pp. 89-124.

[19] Z. Zhou, S. Das, H. Gupta, Connected k-Coverage Problem in Sensor Networks. *Proceedings of the 13th International Conference on Computer Communications and Networks*, 2004, pp. 373 -378.